

Study on Various Methodologies/Frameworks used to Achieve Software Quality in different Organizations

Rachna Lekh and Pooja

ABSTRACT

Purpose of this study is to explain that how the various methodologies, frameworks and processes leads to software quality. Numerous methodologies have been introduced on which lot of exploration or research has been made. This paper includes the study of some processes and techniques which have been used by the firms to achieve software quality. This study strongly recommends the necessity of consciousness regarding the use of software quality oriented processes and the arenas on which extra work leads to quality of software or quality assurance.

Keywords

Software quality, methodologies, frameworks, software quality

1. INTRODUCTION

As per the trend, technology is growing day by day. The sector of IT has been tremendously applied in every type of work area whether it is banking sector, shopping mall, educational sector, restaurants etc. Today the information systems are the need of every arena. Soto manage day to day tasks, information technology has been adopted by the organizations worldwide. As the level of usage of software is increased, the scale of quality is also been moved up. Now the quality of software has become the main requirement of the consumer as well as for the software companies who develop software. Software quality management has become a crucial part for the software development firms. Organizations have to adopt the quality processes, standard formats, quality standards while developing software. Many process improvement standards have been proposed by CMMi, ISO.

As per the survey, author in [2] says that Prince Methodology is one of the process improvement methods which can be adopted to achieve software quality.

2. WHY IT IS REQUIRED TO ADOPT STANDARDS AND PROCESS IMPROVEMENT METHODS??

The crucial or primary need was to achieve software quality so different firms opted different work styles. As per the survey[2], it's been observed that around 75% of softwares' have quality problems, 60 % cost overruns. In inspection, Ebert 1999 [2] showed improvement upto 70% after using software improvement processes and 90% of defect reduction. Yang(2001) [2] specified the usage of project based management concept. Macadam(2002) focused on the need of quality control.

3. SOFTWARE QUALITY STRATEGIES USED BY VARIOUS ORGANIZATIONS

Three aspects which can be followed for quality management are – organization, resources and procedures. One of the models[4] which has been adopted by numerous firms is “PDCA”:

- **P** – Plan the objectives & processes
- **Do** – Implementation of processes
- **Check** – Monitor & Measure processes & report results.
- **Act** – Action to continually improved process performance.

Quantitative measures are also needed to measure the activities which have been performed in development process such as extra man hours which were consumed or spent on extra effort to achieve the quality, no. of times testing rounds have been done etc. These measures can be recorded or tracked via various metrics.

The author [16] represents the harmful and helpful metrics. Harmful metrics refers to Cost per defect, Lines of Code and Technical Debt whereas Function Points and Defect Removal Efficiency observed as helpful metrics

Software quality metrics [8] have been introduced to measure quality such as Internal Metrics [8] , External Metrics and quality metrics are used in collaboration with the Quality Model(ISO/IEC 9126). Following figure represents that how internal & external metrics influences the quality and how they depend upon the former step i.e. External quality depends on internal quality.

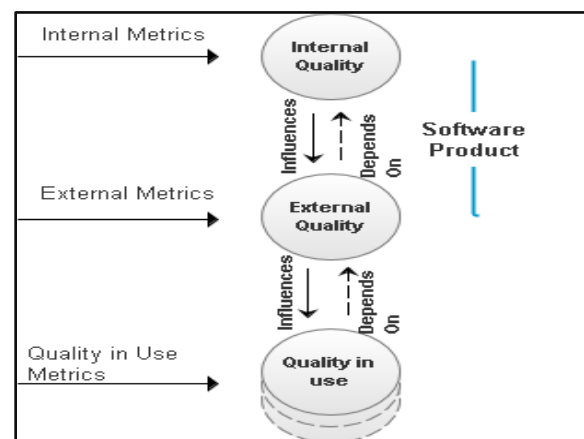


Figure 1 Relationship between types of Metrics

Each & every action which has to be taken must have proper planning as per PRINCE methodology. There should not be any loose end which will become the reason of failure of project. Project manager is a person who must be very intelligent because he will be the key person who has the responsibility of all the project tasks whether it is planning, selection of resources(folks who will be working on the project) on-time delivery of the modules etc.

In PRINCE methodology[2], project has defined a set of products, set of activities, implementation of product, right set of resources, fixed time span. Components of Prince Methodology:

- Product which is required by the customer.
- Action/tasks to construct the products
- Planning criteria which will be required to initiate the product requirements and derived activities.
- Sequence of organization – On the basis of products which accomplish the quality.
- Schedule & costed approach
- An organizational arrangement which defines necessary roles, responsibilities & line of communication and which is connected with the other components.

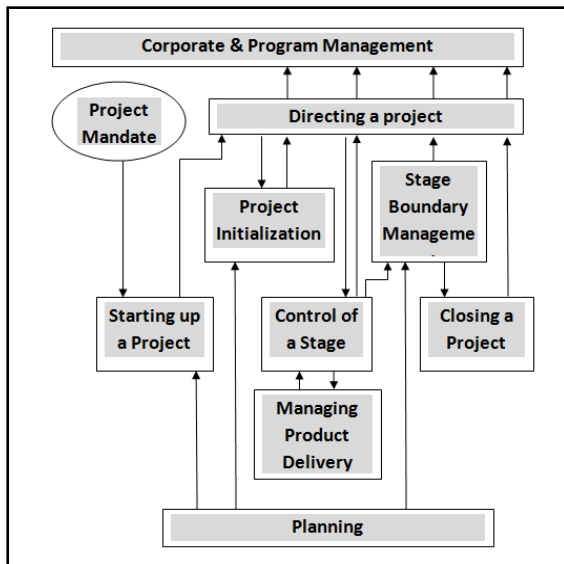


Figure 2 Prince Model

On the other hand, author Nair in another paper stresses on quality assurance or testing processes on quality assurance or testing procedures. Software quality assurance is planned & systematic approach to build a confidence and to make sure that product which has been established in an appropriate product as per the standards.

For quality management systems, ISO 9000 series of standards are specified. Gillies(1997) [2] states that for an organization to achieve a quality process, ISO 9000 possibly will be a good reason.

However team is another important factor in achieving software quality. Domain expert team can reduce no. of chances of defect occurrence in the software product. So PRINCE Methodology alone is not sufficient to achieve software quality.

It measures some of the factors however it is impossible to measure the software quality. Reliability, flexibility, robustness etc are also to be measured for a quality product. It is used by some of the firms who found this methodology in their budget.

Some of the firms say that effective defect management can lead to the software quality [1]. Review process, process audits and proper testing can achieve this goal. Proper management can lead to the success. Author [12] says in effective defect reduction, Project Manager plays important role. Product quality can also be achieved by introducing good practices for identification and fixing of process defects. Effective defect management is stated as crucial need to achieve a quality product. Inspection and testing are two key

methods of effective defect management. Two quality metrics [1] [9] which are introduced - Depth of inspection (DI) and Inspection performance metrics (IPM)[14] [9]. These metrics helps to accomplish the goal of achieving quality software.

Depth of inspection is calculated by assembling all the numerical data along with the testing approaches. Defect capturing ratio could be calculated by this data.

Mathematically,

$$DI = Ni/Td \quad (1)$$

Where DI - Depth of Inspection

Ni – Number of defects by inspection process

Td – Number of defects captured by both inspection & testing

On the other hand, IPM measures the performance people during inspection process.

$$IPM = \text{No. of defects captured by inspection process } (Ni) / \text{Inspection effort } (IE). \quad (2)$$

$$IE = \text{Total No. of Inspectors } (n) * \text{Total amount of inspection time } (T) \quad (3)$$

Total Amount of Inspection time = Actual Inspection Time(It) + Preparation Time(Pi) i = 1 to n indicates number of inspectors

$$IPM = \sum_{i=1} Ni / \sum_{i=1}(Iti + Pti) \quad (4)$$

DI & IPM values can be calculated phase wise or at final stage.

Inspection is one of the efficient quality assurance strategies. Measurement metrics, inspection rate, defect finding efficiency metric are some of the metrics which are required during inspection process.

Inspection Time = Sum of each reviewer’s review time + total person time spent in each meeting.

Inspection detects most of the defects at early stages. Inspection rate is calculated with inspection time and size of artifact in terms of number of pages or LOC i.e. Lines of Code.

Mathematically,

$$\text{Inspection Rate} = \text{Size}/\text{Total Inspection Time}$$

Defect detection rate is estimated as per the efficiency in defect detection.

$$\text{Defect Finding Efficiency} = \text{Total defects found}/\text{Total Inspection Time}$$

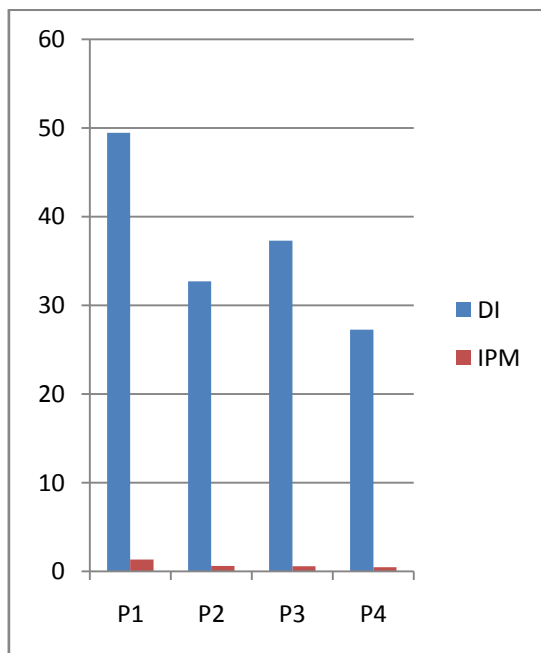
Following table shows the various parameters which represents above metrics and these are applied on real company data.

Table 1. Defect log of few CMMi level company projects to find various parameters

Project Parameters	P1	P2	P3	P4
No. of S/W Personnel	7	10	11	16
Project Development Time	719	1328	1341	4473

No. of Inspectors / Reviewers (N)	2	3	3	4
Total Defects (Td)	190	315	292	946
# Defects found in Inspection process (Ni)	94	103	109	258
Inspection Time(It)	33.2	52	59.3	130.1
Amt. of Inspection Time (T)	35.25	55.5	62.8	136
% Inspection Time	4.6	3.9	4.3	2.9
Inspection Effort (IE)	70.5	166.5	188.4	544
Defect Finding Efficiency	5.75	6.05	4.95	7.27
Depth of Inspection(DI)	49.47	32.69	37.3	27.27
Inspection Performance Metrics(IPM)	1.33	0.61	0.57	0.47

Table 2. Graphical Representation of DI & IPM



DI & IPM values show that how they vary according to Inspection time (%). As the Inspection time (%) decreases consequently DI & IPM decreases.

In order to draw out the non-trivial defects at early stage, inspection must be properly organized in SDLC. This strategy saves cost, augments productivity and quality in terms of rework. So effective defect management primarily includes defect detection.

Testing examines the implemented deliverables whereas inspection examines the deliverables before implementation. In order to get safe from symptoms of the bugs at the later stages of development, inspection recommended as a fruitful method to draw out the defects at early stages. There is less rework by using this process.

Author[13] is conveying to focus on 3Ps –Process, Product & People. This represents that how the use of processes which are used in scheduling, planning , products which are used for development and selection of people who work on the project affects the quality.

On the other hand, author [3] says that small software development firms are also thinking upon better planning, management and software development approaches. However they cannot adopt standards and processes as such. Since these firms are unable to execute these programs due to unavailability of resources, budget and tight deadlines for project completion. So there is new process improvement plan has been made which consists the combination of eXtreme Programming(XP) as a software development method & CMMi v1.2 for development as Software Process Improvement(SPI) model.

Firms would come to know “**what to do for improvement** by SPI” and “**How to do improvement** by software development best practices” with these frameworks[3].

Establishment of SDPIF for SSDFs consists of four stages[3] which are required to be followed.

- Stage 1 – Alignment XP method to CMMi Dev 1.2.
- Stage 2 – Development of proposed SDPIF.
- Stage 3 – Verifying the proposed SDPIF
- Stage 4 – Validating the verified SDPIF.

Stages in detail:

Stage 1 – This stage maps the specific goals of key process areas(KPAs) with the practices of XP method. Three arenas have been made to find the coverage ratio :

- Largely Supported – Twelve KPAs under this : Project planning, project monitoring & control, configuration management, technical solution, product integration, verification, validation, integrated project management + IPPD, risk management, decision analysis and resolution, and quantitative project management, causal analysis & resolution.
- Partially Supported – Eight KPAs under this arena : Requirement management, Measurement & Analysis, Process & product quality assurance, Requirement Development, Organizational Process Definition + IPPD, Organizational, Organizational Process Performance and Organizational Innovation & deployment.
- Not Supported – Organizational Process Focus & Supplier Agreement management.

Stage 2 – Fulfillment of the partially supported and not supported KPAs of CMMi-Dev 1.2 is done in this stage by extending XP method. In this situation, EBA (Extension-based approach) has been adopted.

- P1 – Specifies the requirements and improvements that are required to accomplish the omitted KPAs of CMMi Dev 1.2.

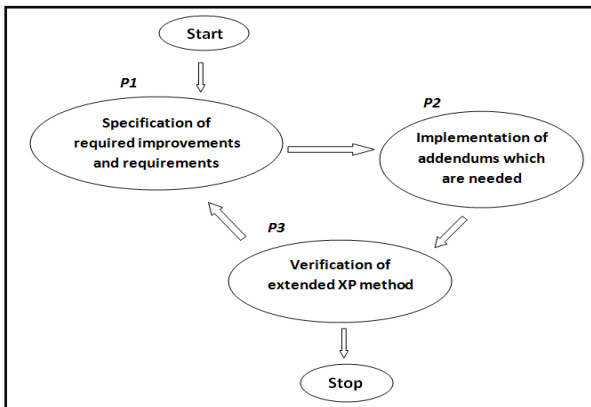


Figure 1 EBA Process

- P2 – Implement and introduce the required addendums such as development models.
- P3 – Verification of the extended XP method if it is applicable for agile method.

Stage 3 – Verification of proposed SDPIF to the specific goals of KPAs of CMMi Dev 1.2 is done in this stage. Focus group members (professional developers & managers with expert researchers) execute focus group rounds for the verification.

Table 3. Activities performed during sessions

Round	Session	Activities
R. 1	S.1	<ul style="list-style-type: none"> • Researcher introducing himself. • Thanking the focus group members. • Presenting the research problem. • Presenting the purpose of the research.
	S.2	<ul style="list-style-type: none"> • Explaining the verification questions. • Answering the verification question individually. • Explaining if there are any inquiries about the verification questions.
	S.3	<ul style="list-style-type: none"> • Discussing the answers and suggestions of each focus group member by all the members.
R.2	S.1	<ul style="list-style-type: none"> • Modifying the proposed framework as suitable suggestions of focus group members.
R.3	S.1	<ul style="list-style-type: none"> • Viewing the verified framework to the members • Asking if there is need for more new modifications.

Stage 4 – Two approaches validate the verified SDPIF:

- Quantitative Research – Suitability of framework is checked by survey in this research.
- Qualitative Research – Under this research, validation of applicability & effectiveness of verified SDPIF for SSDFs is done.

Following table introduced after a survey for the definition of suitability.

Table 4. Interval scale definition of suitability

Interval	Degree of Suitability
From 1 to 1.80	Strongly Unsuitable
From 1.81 to 2.61	Unsuitable
From 2.62 to 3.42	Average
From 3.43 to 4.23	Suitable
From 4.24 to 5	Strongly suitable

There are four generic phases which comprises of Verified extended-XP method.

Phase 1 – Requirement Management

Phase 2 – Development

Phase 3 – Product Delivery along with Product & process efficiency

Phase 4 – Maintenance

All of this concludes that SSDFs ignores the usage of standards and best practices. SDPIF is especially made for SSDFs to let them manage and improve their activities for the execution of software development.

In another paper, author Rahman states [4] about the best practices which have been used and introduced by firms such PDCA method (Plan → Do → Check → Act), Bootstrap, CMMi-ACQ-v1.2, SW-CMM (Software Capability Maturity Model) and many more. Following table represents the various practices.

Table 5. Comparison of process Improvement best Practices – assessment based approach

Best Practice	Levels/Steps/Activities/ Description
SW-CMM	Maturity Levels: <ul style="list-style-type: none"> • Initial • Repeatable • Defined • Managed • Optimizing

CMMI	<p>Staged Approach Maturity Levels:</p> <ul style="list-style-type: none"> • Initial • Repeatable • Defined • Managed • Optimizing <p>Continuous approach:</p> <ul style="list-style-type: none"> • Process • Project • Engineering • Support
Bootstrap	<p>Maturity Levels:</p> <ul style="list-style-type: none"> • Initial • Repeatable • Defined • Managed • Optimizing
ISO/IEC 15504	<p>Process Groups:</p> <ul style="list-style-type: none"> • Organizational Process • Management Process • Customer Supplier Process • Engineering Process • Support Process

Table 6. Comparison of Process Improvement best practices – Non assessment based approach

Best Practice	Levels/Steps/Activities/ Description
PDCA	<p>Steps:</p> <ul style="list-style-type: none"> • Plan • Do • Check • Act
Ishikawa	<p>Steps:</p> <ul style="list-style-type: none"> • Determine goals & targets • Determine the methods of reaching goals • Engage in Education & Training • Implement Work • Check the Effects of Implementation • Take Appropriate Action

QIP	<p>Approach:</p> <ul style="list-style-type: none"> • Characterize & Understand • Set Goals • Choose Processes, Methods, Techniques & Tools • Execute the processes • Analyze results • Package & Store experience
IDEAL Model	<p>Approach:</p> <ul style="list-style-type: none"> • Initiating • Diagnosing • Establishing • Acting • Leveraging
ISO 15504	<p>Approach:</p> <ul style="list-style-type: none"> • Examine Organization Needs • Initiate Process Improvement • Prepare & Conduct Process Assessment • Analyze results& derive action plan • Implement Improvements • Confirm Improvements • Sustain Improvement gains • Monitor performance

4. CONCLUSION

This paper represents the methodologies and best practices used by various firms in order to achieve goals and planned results within a defined schedule and a budget. Firms choose methodologies to use as per the resource and budget availability. Even after a numerous studies and number of methodologies, We have not yet have a standard methodology which benefits small as well as big firms equally and that too keeping the resource and budget constraints answered. By means of this Paper , We want to present a model as a solution to wards this standard methodology by simply identifying the best combination of the sub processes which will lead small as well as big firms towards their goal

5. REFERENCES

- [1] T.R. Gopalakrishnan Nair, V. Suma, P. Kumar Tiwari, "Significance of depth inspection & inspection performance metrics for consistent defect management in software industry", The Institute of Engineering & Technology (IET), Vol. 6, Issue 6, pp 524-535,(2012)
- [2] Shubo XU, Dishu Xu, "Project Management methodologies": Are they sufficient to develop quality software", IEEE(2011)
- [3] Mejhem Yousef Al – Tarawneh, Mohd. Syazwan Abdullah & Jasem Alostad, "Software Development Process Improvement Framework(SDPIFs) for small software development firms(SSDFs)" , International

- Journal of Computer Science Issues (IJCSI), Vol.10, Issue 1, (2013).
- [4] Aedah Abd Rahman, Shamsul Sahibuddin & Suhaini Ibrahim, “A Study of Process Improvement Best Practices”, IEEE(2011).
- [5] Changli Sun, “Software Document quality Measurement – A Fuzzy approach” , IEEE(2010).
- [6] Li Wei, lin Xiao , Yu Wuyi, Yin Zhao, Cai, Jianhui & Li Maoqing, “The Research Appliance of Multi Layer Fuzzy Comprehensive evaluation in the Appraisal of software quality”, IEEE(2008).
- [7] Ahmad Javan Bakht, Mohsen Fallah Rad & Farshad Akbari, “Using Fuzzy multiple criteria decision making in evaluation of software quality”, IEEE(2010).
- [8] ISO/IEC 9126-3: Software Engineering – Product Quality – Part 3 : Internal Metrics,2002
- [9] Nithya G. Nair, Suma V, T.R. Gopalakrishnan, “Estimation of Characteristics of software team for implementing effective process through Inspection Performance Metrics”.
- [10] Bob McFeeley, “A User Guide for Software Process Improvement”,Software Engineering Institute (SEI), Feb 1996.
- [11] ISO/IEC 9126-1: 2001, Software Engineering-Product Quality-Part 1:Quality Model
- [12] N.R. Shashi Kumar, T.R. Gopalakrishnan, V. Suma, “An Analytical Approach for project Manager in Effective Defect Management in Software Process”, IEEE, 2011.
- [13] Poomima U.S, Suma V, “Significance of Quality Metrics during Software Development Process”
- [14] V. Suma, T.R. Gopalakrishnan, “”Defect Management using DI & IPM.
- [15] Tom Kendrick, “”Defining & Implementing Metrics for Project Risk Reduction”.
- [16] Caper Jones, “Three Harmful Metrics & Two Helpful Metrics”, 2012
- [17] Topi Haapio, “ Improving Effort Management in Software Development Projects”, Publications of University of eastern Finland, 2011.
- [18] Sylvie Trudel, Jean Marc Lavoie, Marie-Claude Pare, Witold Suryn, “PEM: The Small Company Dedicated Software Process Quality Evaluation Method combining CMMI & ISO/IEC 14598, Springer, 2006”